WEST Search History

| Hide Items | Restore | Clear | Cancel |

DATE: Monday, April 19, 2004

Hide?	Set Name	Query	Hit Count
		DB=EPAB,JPAB,DWPI,TDBD; PLUR=NO; OP=ADJ	
☐	L30	shared memor$3 and L29	3
☐	L29	matrix and diagonal and block$1	134
☐	L28	L27 and diagonal blocks	3
☐	L27	lu or lower upper	1253521
☐	L26	matrix blocks and diagonal blocks	0
		DB=PGPB,USPT; PLUR=NO; OP=ADJ	
☐	L25	(lu or lower upper) and L21	3
☐	L24	(lu or lower upper) with L21	0
☐	L23	L1 and L22	0
☐	L22	L12 and L21	3
☐	L21	(712/17).ccls.	67
☐	L20	L12 and L19	0
☐	L19	(708/514).ccls.	16
☐	L18	L12 and L17	6
☐	L17	(708/520).ccls.	142
☐	L16	matrix blocks same diagonal blocks	3
☐	L15	matrix blocks with diagonal blocks	2
☐	L14	matrix with diagonal blocks	36
☐	L13	matrix with diagonal with block$1	341
☐	L12	matrix same diagonal same block$1	1955
☐	L11	L1 and vector processor$1	1
☐	L10	L1 and array processor$1	1
☐	L9	L1 and L8	4
☐	L8	parallel computer$1	2307
☐	L7	L1 and L6	5
☐	L6	shared memor$3	10121
☐	L5	diagonal same L4	0
☐	L4	lower upper factorization	3
☐	L3	diagonal same L2	8
☐	L2	matrix same L1	39
☐	L1	lu factorization	53

WEST Search History

Hide Items | Restore | Clear | Cancel

DATE: Monday, April 19, 2004

Hide?	Set Name	Query	Hit Count
		DB=PGPB,USPT; PLUR=NO; OP=ADJ	
☐	L25	(lu or lower upper) and l21	3
☐	L24	(lu or lower upper) with l21	0
☐	L23	l1 and L22	0
☐	L22	l12 and L21	3
☐	L21	712/17.ccls.	67
☐	L20	l12 and L19	0
☐	L19	708/514.ccls.	16
☐	L18	l12 and L17	6
☐	L17	708/520.ccls.	142
☐	L16	matrix blocks same diagonal blocks	3
☐	L15	matrix blocks with diagonal blocks	2
☐	L14	matrix with diagonal blocks	36
☐	L13	matrix with diagonal with block$1	341
☐	L12	matrix same diagonal same block$1	1955
☐	L11	l1 and vector processor$1	1
☐	L10	l1 and array processor$1	1
☐	L9	l1 and L8	4
☐	L8	parallel computer$1	2307
☐	L7	l1 and L6	5
☐	L6	shared memor$3	10121
☐	L5	diagonal same L4	0
☐	L4	lower upper factorization	3
☐	L3	diagonal same L2	8
☐	L2	matrix same L1	39
☐	L1	lu factorization	53

END OF SEARCH HISTORY

WEST Search History

| Hide Items | Restore | Clear | Cancel |

DATE: Monday, April 19, 2004

Hide?	Set Name	Query	Hit Count
		DB=PGPB,USPT; PLUR=NO; OP=ADJ	
☐	L20	l12 and L19	0
☐	L19	708/514.ccls.	16
☐	L18	l12 and L17	6
☐	L17	708/520.ccls.	142
☐	L16	matrix blocks same diagonal blocks	3
☐	L15	matrix blocks with diagonal blocks	2
☐	L14	matrix with diagonal blocks	36
☐	L13	matrix with diagonal with block$1	341
☐	L12	matrix same diagonal same block$1	1955
☐	L11	l1 and vector processor$1	1
☐	L10	l1 and array processor$1	1
☐	L9	l1 and L8	4
☐	L8	parallel computer$1	2307
☐	L7	l1 and L6	5
☐	L6	shared memor$3	10121
☐	L5	diagonal same L4	0
☐	L4	lower upper factorization	3
☐	L3	diagonal same L2	8
☐	L2	matrix same L1	39
☐	L1	lu factorization	53

END OF SEARCH HISTORY

☐ | Generate Collection | | Print |

L3: Entry 6 of 8   File: USPT   Feb 18, 1997

US-PAT-NO: 5604911
DOCUMENT-IDENTIFIER: US 5604911 A

TITLE: Method of and apparatus for preconditioning of a coefficient matrix of
simultaneous linear equations

DATE-ISSUED: February 18, 1997

INVENTOR-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY
Ushiro; Yasunori	Yokosuka			JP

ASSIGNEE-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY	TYPE CODE
Hitachi, Ltd.	Tokyo			JP	03

APPL-NO: 07/ 947801   [PALM]
DATE FILED: September 21, 1992

FOREIGN-APPL-PRIORITY-DATA:

COUNTRY	APPL-NO	APPL-DATE
JP	3-239260	September 19, 1991

INT-CL: [06] G06 F 15/347

US-CL-ISSUED: 395/800; 364/578
US-CL-CURRENT: 703/2

FIELD-OF-SEARCH: 395/800, 364/578

PRIOR-ART-DISCLOSED:

U.S. PATENT DOCUMENTS

| Search Selected | | Search ALL | | Clear |

	PAT-NO	ISSUE-DATE	PATENTEE-NAME	US-CL
☐	4697247	September 1987	Grinberg et al.	364/713
☐	4787057	November 1988	Hammond	364/754
☐	5301342	April 1994	Scott	395/800

OTHER PUBLICATIONS

Chen et al., "Parallel LU Factorization for Circuit Simulation on a MIMD Computer," International Conference on Computer Design, pp. 129-132. May 1988.
Bratkovic, "Improved Numerical Stability of Sparse Matrix Reduction Method," IEEE International Symposium on Circuits and Systems, pp. 631-634. Aug. 1988.
Gustafsson, "A Class of First Order Factorization Methods," BIT 18 (1978), pp. 142-156.
Van Der Vorst, "ICCG and Related Methods for 3D Problems on Vector Computers," Comp. Phys. Communications, 53 (1989), pp. 223-235.
Van Der Vorst, "B1-CGSTAB: A First and Smoothly Converging Variant of BI-CG for the Solution of Nonsymmetric Linear Systems," pp. 1-16 - Could Not Be Considered Due to Lack of Publishing Date.
Ma et al., "2 Iterative Methods . . . 3 Vectorization and Parallelization," vol. 4, Winter 1990, pp. 12-24.

ART-UNIT: 232

PRIMARY-EXAMINER: Bowler; Alyssa H.

ASSISTANT-EXAMINER: Harrity; John

ATTY-AGENT-FIRM: Antonelli, Terry, Stout & Kraus, LLP

ABSTRACT:

To analyze a physical phenomenon by a computer having a plurality of vector processors and a parallel computer, there is generated submatrices in a preconditioning for obtaining solutions of simultaneous linear equations. Nonzero elements of the coefficient matrix are stored with column number indices assigned thereto such that the elements of the coefficient matrix and the data of right-side vector are scaled according to a sum of absolute values of nondiagonal elements of the coefficient matrix and a diagonal element related thereto. The nonzero elements are sorted depending on magnitude of their absolute values to subdivide the nondiagonal nonzero elements into m submatrices E1, E2, . . . , Em each having substantially a comparable order. Using products developed between differences between a unit matrix and these submatrices in the iterative calculations for a large-sized numerical simulation, there is obtained quite a satisfactory characteristic of convergence of solutions and hence the processing speed is remarkably increased.

15 Claims, 14 Drawing figures

☑ ▓▓▓ Generate Collection ▓▓▓ | Print |

L3: Entry 6 of 8   File: USPT   Feb 18, 1997

DOCUMENT-IDENTIFIER: US 5604911 A
TITLE: Method of and apparatus for preconditioning of a coefficient matrix of
simultaneous linear equations

Brief Summary Text (19):
First, to establish the calculation method suitable for the parallel processing,
the incomplete LU factorization of the matrix A is not achieved in the
preconditioning. Namely, the non-diagonal nonzero elements of the matrix A are
subdivided into m submatrices E1, E2, . . . , Em such that each preconditioning
matrix w is formed with multiplications between (I-Ei), i=1, 2, . . . , m.
Resultantly, in the overall region of the vector operations for the solution of
linear equations, there can be obtained a degree of parallelization almost
identical to the order n of the matrix A
(n=n.sub.x .multidot.n.sub.y .multidot.n.sub.z for a three-dimensional matrix and
n=n.sub.x .multidot.n.sub.y for a two-dimensional matrix).
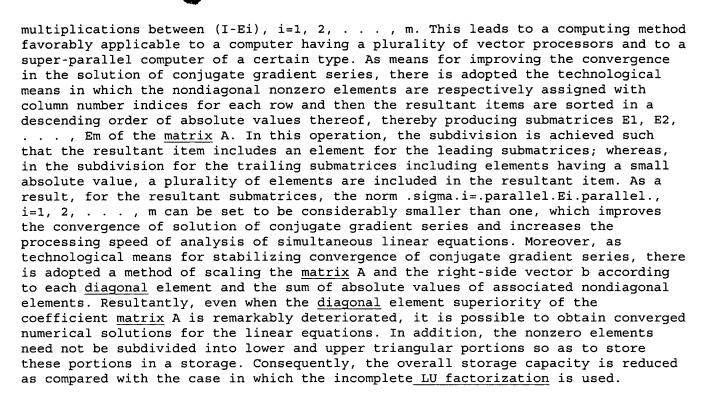
Brief Summary Text (26):
In the conventional method, the incomplete LU factorization is directly conducted
on the coefficient matrix A. As contrast thereto, according to the present
invention, the absolute value of each diagonal element is combined with the sum of
absolute values of associated nondiagonal elements. Values resultant from the
operation achieved above are assumed to be diagonal elements in the incomplete LU
factorization. In this method, the correction coefficient .alpha. can be set to a
fixed value in the Gustafsson-type correction, which is not remarkably influenced
from the deterioration of the property of the coefficient matrix thus attained,
thereby keeping the stability of the convergence.

Brief Summary Text (27):
Furthermore, in a case where the incomplete LU factorization is adopted in the
preconditioning, even when the coefficient matrix is in an ill condition, namely,
has an inappropriate property (the sum of absolute values of nondiagonal elements
is larger than the absolute value of the associated diagonal element), there are
developed an advantage of a satisfactory stability of convergence and a
disadvantage of deterioration in the processing efficiency of the plural vector
processing units. On the other hand, using the matrix constituted with
multiplications between (I-Ai), i=1, 2, . . . , m of the second aspect in the
preconditioning, there are obtained reversed results of the advantage and
disadvantage associated with the incomplete LU factorization. Consequently, as
technological means for developing a high-speed stable convergence under any
conditions, there is adopted a method in which the number of available vector
processors and a ratio between an absolute value of each diagonal element and the
sum of absolute values of nondiagonal elements related thereto are employed as
parameters to select a favorable one of two preconditioning procedures.

Brief Summary Text (28):
As the preconditioning for the solution of conjugate gradient series, although the
processing of the incomplete LU factorization can be established to be suitable for
vector processors, it is difficult to set the processing to be oriented to the
super-parallel processing. According to the first aspect of the present invention,
the preconditioning matrix is configured with products resultant from

multiplications between (I-Ei), i=1, 2, . . . , m. This leads to a computing method favorably applicable to a computer having a plurality of vector processors and to a super-parallel computer of a certain type. As means for improving the convergence in the solution of conjugate gradient series, there is adopted the technological means in which the nondiagonal nonzero elements are respectively assigned with column number indices for each row and then the resultant items are sorted in a descending order of absolute values thereof, thereby producing submatrices E1, E2, . . . , Em of the matrix A. In this operation, the subdivision is achieved such that the resultant item includes an element for the leading submatrices; whereas, in the subdivision for the trailing submatrices including elements having a small absolute value, a plurality of elements are included in the resultant item. As a result, for the resultant submatrices, the norm .sigma.i=.parallel.Ei.parallel., i=1, 2, . . . , m can be set to be considerably smaller than one, which improves the convergence of solution of conjugate gradient series and increases the processing speed of analysis of simultaneous linear equations. Moreover, as technological means for stabilizing convergence of conjugate gradient series, there is adopted a method of scaling the matrix A and the right-side vector b according to each diagonal element and the sum of absolute values of associated nondiagonal elements. Resultantly, even when the diagonal element superiority of the coefficient matrix A is remarkably deteriorated, it is possible to obtain converged numerical solutions for the linear equations. In addition, the nonzero elements need not be subdivided into lower and upper triangular portions so as to store these portions in a storage. Consequently, the overall storage capacity is reduced as compared with the case in which the incomplete LU factorization is used.

Brief Summary Text (29):
As the preconditioning for solution of conjugate gradient series, although the processing procedure can be established to be suitable for vector processors according to the incomplete LU factorization, it is difficult to set the processing to be appropriate for the super-parallel processing. According to the second aspect of the present invention, the preconditioning matrix is structured with products resulted from multiplications between (I-Ai), i=1, 2, . . . , m, thereby implementing a calculation method favorably applicable to a computer having a plurality of vector processors and a super-parallel computer. Moreover, by configuring the preconditioning matrix with the results from multiplications between (I-Ai), i=1, 2, . . . , m, as compared with the conventional case where a formula E=A.sub.1 +A.sub.2 + . . . +Am is adopted to establish a formula I-E+E.sup.2 -E.sup.3 +E.sup.4 . . . , the convergence of solution can be improved for the gradient series. In addition, as technological means for stabilizing convergence of solution of conjugate gradient series, there is adopted a method of scaling the matrix A and the right-side vector b according to each diagonal element and the sum of absolute values of associated nondiagonal elements. Resultantly, even when the diagonal element superiority of the coefficient matrix A is remarkable deteriorated, it is possible to obtain converged numerical solutions for the linear equations in a stable condition.

Brief Summary Text (30):
In the conventional method in which the incomplete LU factorization is directly conducted on the coefficient matrix A, when the diagonal element superiority is considerably deteriorated, there appears a drawback that the conjugate gradient series with preconditioning does not lead to the convergence of solution. According to the third aspect of the present invention, the values generated by combining the absolute value of each diagonal element with the sum of absolute values of associated nondiagonal elements are assumed to be diagonal elements in the incomplete LU factorization. It is therefore possible to avoid the disadvantageous state in which the diagonal elements after factorization are extremely minimized during the incomplete LU factorization. This leads to an advantageous effect of a stable convergence even when the diagonal element superiority of the coefficient matrix A is remarkably deteriorated. Moreover, there is also attained an advantage that the convergence speed is rarely changed even when the correction

coefficient .alpha. of the Gustafsson-type correction is fixed to about 0.95. Furthermore, when conducting iterative calculations for solution of linear equations according to conjugate gradient series in a parallel computer having a plurality of vector processors, the method in which the incomplete LU factorization is adopted in the preconditioning and the method in which the matrix structured with products resultant from multiplications between (I-Ai), i=1, 2, . . . , m are used in the preconditioning are respectively attended with advantageous and disadvantageous features. Such features are strongly influenced from two parameters including the number of available vector processors and the ratio between the absolute value of diagonal element and the sum of absolute values of nondiagonal elements. Consequently, depending on these parameters, either one of the two preconditioning procedures suitable for the given condition is automatically selected to achieve calculations at a higher speed.

Detailed Description Text (28):
In a step 51 of FIG. 10, there is shown an example of a preparative operation for conducting an incomplete LU factorization on the coefficient matrix A. Namely, the sum .mu.i of absolute values of nondiagonal elements is computed for each row. There is also calculated a value wi to be assumed as a diagonal element in the incomplete LU factorization. In the formulae, N stands for the dimensionality of the matrix and the symbols other than .mu.i and wi are associated with those shown in FIG. 11. Symbols a.sub.i, b.sub.i, c.sub.i, e.sub.i, f.sub.i, and g.sub.i denotes nondiagonal elements. The value of di is set to be greater than zero (di>0). In a step 52, there is conducted an incomplete LU factorization corrected according to the correction proposed by Gustaffson. In this computation, d.sub.i =1/{wi- . . . } is employed in place of the conventional formula d.sub.i =1/{di- . . . }. Resultantly, even when the coefficient matrix is ill conditioned (for a diagonal element, the sum of absolute values of associated nondiagonal is greater than the absolute value of the diagonal element), the convergence of solution is stabilized. Moreover, regardless of the ill-conditioned matrix, the value of the correction coefficient .alpha. can be fixed to about 0.95. In addition, to increase the convergence speed, the value of .alpha. need only be increased as compared with the dimensionality or order N, where .alpha. is less than one. Furthermore, a.sub.i and g.sub.i, b.sub.i and f.sub.i, and c.sub.i and e.sub.i are arranged at positions apart from the diagonal positions by m, one, and one, respectively. When a vector computer is used for the computation, the calculation of the step 52 need not be achieved in the order of i=1, 2, 3, . . . , N. It is only necessary to order the items in a direction according to the hyper-plane method. This technology has already been broadly known.

Detailed Description Text (29):
A reference numeral 53 of FIG. 11 denotes the configuration indicating positions of nonzero elements of the original coefficient matrix A. In this graph, a letter d stands for a diagonal matrix, letters a, b, c, e, f, and g designate non diagonal matrices, and letters a.sub.i, b.sub.i, c.sub.i, d.sub.i, e.sub.i, f.sub.i, and g.sub.i indicate elements of row i in the respective matrices. A graph 54 shows positional configuration of nonzero elements of the lower triangular matrix L attained from the incomplete LU factorization, a graph 55 presents the positional structure of diagonal matrix D, and a graph 56 indicates a positional constitution of nonzero elements of the upper triangular matrix U. In these graphs, each of the symbols a, b, etc. indicate a matrix of the same value as the associated matrix shown in the graph 53 or 54. Only the value of d is additionally calculated.

Detailed Description Text (32):
FIG. 13 is a flowchart showing the operation of selecting a preconditioning method in which submatrices are generated for KEY=0 and the incomplete LU factorization (LDU) is employed as the preconditioning for KEY=1. In a step 61, S is set to the maximum value of the ratio between the value of a diagonal element di and the sum of absolute values of nondiagonal elements associated therewith. If S takes a small value, the property of the matrix is satisfactory; otherwise, the matrix is ill

conditioned. In a decision step 62, when the coefficient <u>matrix</u> A is completely a <u>diagonal</u> element superior <u>matrix</u> in the non-stationary computation or the like, namely, for S.ltoreq.1-.epsilon..sub.1, it is decided to set 0 to KEY (KEY=0). In general, .epsilon..sub.1 is set to about 0.1. In a step 63, when the <u>diagonal</u> element superiority of the coefficient <u>matrix</u> A is remarkably deteriorated, namely, for S.gtoreq..epsilon..sub.2, it is decided to set one to KEY (KEY=1). In general, .epsilon..sub.2 is set to about two. In a step 64, it is judged, when the coefficient <u>matrix</u> A is found to be not particularly satisfactory or unsatisfactory, to determine whether or not a plurality of vector processors can be adopted for the computation. If this is the case, KEY is set to 0. If only one vector processor is available, KEY is set to one. For KEY=0, there is selected a preconditioning method suitable for the plural vector processors. Namely, the vector processors develops a highly efficient operation.

<u>Detailed Description Text</u> (35):
According to the embodiment, using the iterative solution of conjugate gradient series with incomplete <u>LU factorization,</u> there is obtained an advantage of a stable computation of numerical solutions for simultaneous linear equations. Particularly, it is possible to avoid the conventional case in which when the <u>diagonal</u> superiority is greatly deteriorated for the <u>matrix,</u> the converged solution cannot be attained according to the conventional incomplete <u>LU factorization</u>. Moreover, by setting the correction coefficient .alpha. of the Gustafsson-type correction is fixed to be about 0.95 regardless of the property of <u>diagonal</u> element superiority of the <u>matrix,</u> the characteristic of convergence can be improved to be three to five times, as compared with the conventional case, for a two-dimensional problem of 100 by 100 subdivision. Furthermore, when the value of .alpha. is increased in proportion to the subdivision number used to subdivide the pertinent area (.alpha.<1.0), the convergence speed can be much more increased.

CLAIMS:

7. An apparatus according to claim 1, further including:

means for achieving an incomplete <u>LU factorization</u> for the coefficient matrix of said linear equations by decomposing the coefficient <u>matrix</u> into a product of a lower triangular <u>matrix</u> L and an upper triangular <u>matrix</u> U in accordance with <u>diagonal</u> elements thereof; and

means for determining whether generating the submatrices or achieving the incomplete LU factorization in accordance with a maximum value of a ratio between an absolute value of a diagonal element on a main diagonal axis and a sum of absolute values of the nondiagonal elements for each row.

10. A computer for use with simultaneous linear equations for computing incomplete LU factorization matrix for a coefficient matrix of the linear equations, wherein the whole coefficient matrix is decomposed into a product of a lower triangular matrix L and an upper triangular matrix U, said computer comprising:

means for storing therein the coefficient matrix, a result of the calculation, and intermediate results thereof; and

calculating and processing means for achieving an incomplete <u>LU factorization</u> of the coefficient <u>matrix</u> by decomposing into a product of lower and upper triangular matrices by reference to contents of the storing means according to a value determined by both a <u>diagonal</u> element on a main <u>diagonal</u> axis and a sum of absolute values of nondiagonal elements for each row.

12. A computer according to claim 10, wherein whether the calculation to attain a preconditioning <u>matrix</u> or the calculation to obtain the incomplete <u>LU factorization</u> <u>matrix</u> is to be executed is determined in accordance with two parameters including

a number of available vector processors and a ratio between a sum of absolute values of nondiagonal elements of the coefficient matrix and a value of a diagonal element related thereto.

☐ ▐ Generate Collection ▌ ▐ Print ▌

```
L3: Entry 5 of 8                    File: USPT                    Nov 7, 2000
```

US-PAT-NO: 6144932
DOCUMENT-IDENTIFIER: US 6144932 A

TITLE: Simulation device and its method for simulating operation of large-scale electronic circuit by parallel processing

DATE-ISSUED: November 7, 2000

INVENTOR-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY
Hachiya; Koutarou	Tokyo			JP

ASSIGNEE-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY	TYPE CODE
NEC Corporation	Tokyo			JP	03

APPL-NO: 09/ 089163   [PALM]
DATE FILED: June 2, 1998

FOREIGN-APPL-PRIORITY-DATA:

COUNTRY	APPL-NO	APPL-DATE
JP	9-143955	June 2, 1997

INT-CL: [07] G06 F 17/50

US-CL-ISSUED: 703/14; 703/15, 703/16, 708/446
US-CL-CURRENT: 703/14; 703/15, 703/16, 708/446

FIELD-OF-SEARCH: 395/500.34, 395/500.35, 395/500.36, 395/500.37, 395/500.23, 708/446, 703/13, 703/14, 703/15, 703/16, 703/2

PRIOR-ART-DISCLOSED:

U.S. PATENT DOCUMENTS

▐ Search Selected ▌ ▐ Search ALL ▌ ▐ Clear ▌

	PAT-NO	ISSUE-DATE	PATENTEE-NAME	US-CL
☐	5313398	May 1994	Rohrer et al.	395/500.35
☐	5369594	November 1994	Huange et al.	395/500.35
☐	5442569	August 1995	Osano	395/500.21
☐	5490278	February 1996	Mochizuki	708/446

☐ 5754181   May 1998   Amdursky et al.   345/419

☐ 5768160   June 1998   Kakegawa   395/500.37

OTHER PUBLICATIONS

Eickhoff et al., "Levelized Incomplete LU Factorization and its Application to
Large-Scale Circuit Simulation", IEEE Transactions on Computer-Aided Design of
Integrated Circuits and Systems, vol. 14, Issue 6, pp. 720-727, Jun. 1995.
Mayaram et al., "A Parallel Block-Diagonal Preconditioned Conjugate-Gradient
Solution Algorithm for Circuit and Device Simulationts", Digest of Technical Papers
from the 1990 IEEE International Conference on Computer-Aided Design, pp. 446-449,
Nov. 1990.
Wu et al., "Performance Tuning of a Multiprocessor Sparse Matrix Equation Solver",
Proceedings of the Twenty-Eighth Hawaii International Conference on System
Sciences, vol. II, pp. 4-13, Jan. 1995.
Simic et al., "Partitioning Strategies within a Distributed Multilevel Logic
Simulaotr including Dynamic Repartitioning", Proceedings of the European Design
Automation Conference, pp. 96-101, Sep. 1993.
Burch et al., "A new Matrix Solution Technique for General Circuit Simulation",
IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, vol.
12, Issue 2, pp. 225-241, Feb. 1993.
Mayaram et al., "A Parallel Block-Diagonal Preconditioned Conjugate-Gradient
Solution Algorithm for Circuit And Device Simulations", pp. 446-449, (1990).
Mayaram et al., "A Parallel Block-Diagonal Preconditioned Conjugate-Gradient
Solution Algorithm for Circuit And Device Simulations", pp. 446-449, (1990).

ART-UNIT: 273

PRIMARY-EXAMINER: Stamber; Eric W.

ASSISTANT-EXAMINER: Sergent; Douglas W.

ATTY-AGENT-FIRM: Foley & Lardner

ABSTRACT:

A simulation device comprises an equation generating unit for generating a
simultaneous linear equation by application of the implicit integration formula and
the Newton iteration method to the description data of an electronic circuit to be
simulated, a plurality of block ILU factorization units for performing incomplete
LU factorization processing in parallel on each block in a coefficient matrix of
the generated simultaneous linear equation, a plurality of fill-in adding units for
adding a plurality of fills-in generated by the incomplete LU factorization to a
combined portion of coefficient matrices, in parallel, a plurality of line
collection ILU factorization units for ILU-factorizing each of several line
collections on the combined portion where the fills-in are added, and a convergent
solution judging unit for repeating a series of the above processing until
convergence of a solution in the simultaneous linear equation generated by the
equation generating unit is reached.

8 Claims, 14 Drawing figures

☐ ▮ Generate Collection ▮ ▮ Print ▮

L3: Entry 5 of 8   File: USPT   Nov 7, 2000

DOCUMENT-IDENTIFIER: US 6144932 A
TITLE: Simulation device and its method for simulating operation of large-scale
electronic circuit by parallel processing

Brief Summary Text (21):
a plurality of block ILU factorization means for performing incomplete LU
factorization processing in parallel on each of several blocks in a bordered-block-
diagonal coefficient matrix of the simultaneous linear equation generated by the
equation generating means,

Brief Summary Text (38):
a plurality of block ILU factorization means for performing incomplete LU
factorization processing in parallel on each of several blocks in a bordered-block-
diagonal coefficient matrix of the simultaneous linear equation generated by the
equation generating means,

Brief Summary Text (50):
a step of performing incomplete LU factorization processing in parallel on each of
several blocks in a bordered-block-diagonal coefficient matrix of the simultaneous
linear equation generated by the equation generating step,

Brief Summary Text (65):
a step of performing incomplete LU factorization processing in parallel on each of
several blocks in a bordered-block-diagonal coefficient matrix of the simultaneous
linear equation generated by the equation generating step,

Brief Summary Text (75):
a step of performing incomplete LU factorization processing in parallel on each of
several blocks in a bordered-block-diagonal coefficient matrix of the simultaneous
linear equation generated by the equation generating step,

Brief Summary Text (89):
a step of performing incomplete LU factorization processing in parallel on each of
several blocks in a bordered-block-diagonal coefficient matrix of the simultaneous
linear equation generated by the equation generating step,

CLAIMS:

1. A simulation device for simulating an operation of an electronic circuit,
comprising:

a data input means for receiving description data of an electronic circuit to be
simulated;

a data partitioning means for partitioning the description data of the electronic
circuit received from said data input means to generate description data of partial
circuits;

a point deciding means for deciding a time point of the next stage for the
description data of the partial circuits generated by said data partitioning means;

an equation generating means for generating a simultaneous linear equation by application of an implicit integration formula and a Newton iteration method to the description data of the partial circuits at the time point of the next stage as decided by said point deciding means;

a plurality of block ILU factorization means for performing incomplete LU factorization processing in parallel on each of several blocks in a bordered-block-diagonal coefficient matrix of the simultaneous linear equation generated by said equation generating means to produce a plurality of fill-ins, respectively;

a plurality of fill-in adding means for respectively adding a fill-in generated by the corresponding ILU factorization means to a combined portion of coefficient matrices, in parallel;

a plurality of line collection ILU factorization means for respectively ILU-factorizing a corresponding one of several line collections on the combined portion, in parallel;

a convergent solution judging means for repeating a series of processing by said equation generating means, said block ILU factorization means, said fill-in adding means, and said line collection ILU factorization means until convergence of a solution in the simultaneous linear equation generated by said equation generating means is reached;

an operation repeating means for repeating a series of iteration processing by said equation generating means, said block ILU factorization means, said fill-in adding means, said line collection ILU factorization means, and said convergent solution judging means at the time point decided by said point deciding means until the time point reaches a predetermined final time; and

an output means for supplying a series of convergent solutions representing the operation of the electronic circuit over time.

3. A simulation device for simulating an operation of an electronic circuit, comprising:

a plurality of data processing means for executing a simulation of an electronic circuit operation, in various methods different in processing speed and convergence capacity of a solution, by use of an implicit integration formula and a Newton iteration method;

a processing selecting means for selecting a data processing means having the highest processing speed and the lowest convergence capacity, of the plurality of data processing means, so to perform a simulation;

a time-detecting means for detecting iteration times by the Newton iteration method in the processing of said data processing means which performed the simulation; and

a processing switching means for controlling said processing selecting means so as to sequentially switch said data processing means to that one having the higher processing speed and the lower convergence capacity next to said data processing means which performed the simulation, when the iteration times detected by said time-detecting means is beyond a predetermined allowed time;

wherein one of said plurality of data processing means further comprises

a data input means for receiving description data of an electronic circuit to be

simulated,

a data partitioning means for partitioning the description data of the electronic circuit received from said data input means to generate description data of partial circuits,

a point deciding means for deciding a time point of the next stage for the description data of the partial circuits generated by said data partitioning means,

an equation generating means for generating a simultaneous linear equation by application of the implicit integration formula and the Newton iteration method to the description data of the partial circuits at the time point of the next stage as decided by said point deciding means,

a plurality of block ILU factorization means for performing incomplete LU factorization processing in parallel on each of several blocks in a bordered-block-diagonal coefficient matrix of the simultaneous linear equation generated by said equation generating means to produce a plurality of fill-ins, respectively,

a plurality of fill-in adding means for respectively adding a fill-in generated by the corresponding ILU factorization means to a combined portion of coefficient matrices, in parallel,

a plurality of line collection ILU factorization means for respectively ILU-factorizing a corresponding one of several line collections on the combined portion, in parallel,

a convergent solution judging means for repeating a series of processing by said equation generating means, said block ILU factorization means, said fill-in adding means, and said line collection ILU factorization means until convergence of a solution in the simultaneous linear equation generated by said equation generating means is reached,

an operation repeating means for repeating a series of iteration processing by said equation generating means, said block ILU factorization means, said fill-in adding means, said line collection ILU factorization means, and said convergent solution judging means at the time point decided by said point deciding means until the time point reaches a predetermined final time, and

an output means for supplying a series of convergent solutions representing the operation of the electronic circuit over time.

5. A simulation method for simulating an operation of an electronic circuit, comprising the steps of:

a step of receiving description data of an electronic circuit to be simulated;

a step of partitioning the description data of the electronic circuit to generate description data of partial circuits;

a step of deciding a time point of the next stage for the description data of the partial circuits generated by said data partitioning step;
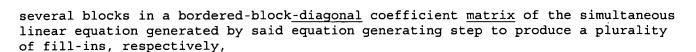
a step of generating a simultaneous linear equation by application of an implicit integration formula and a Newton iteration method to the description data of the partial circuits at the time point of the next stage as decided by said point deciding step;

a step of performing incomplete LU factorization processing in parallel on each of

several blocks in a bordered-block-diagonal coefficient matrix of the simultaneous linear equation generated by said equation generating step to produce a plurality of fill-ins, respectively;

a step of respectively adding a fill-in generated by the corresponding ILU factorization means to a combined portion of coefficient matrices, in parallel;

a step of respectively ILU-factorizing a corresponding one of several line collections on the combined portion, in parallel;

a step of repeating a series of processing by said equation generating means, said block ILU factorization means, said fill-in adding means, and said line collection ILU factorization means until convergence of a solution in the simultaneous linear equation generated by said equation generating means is reached;

a step of repeating a series of iteration processing by said equation generating step, said block ILU factorization step, said fill-in adding step, said line collection ILU factorization step, and said convergent solution judging step at the time point decided by said point deciding step until the time point reaches a predetermined final time; and

a step of outputting a series of convergent solutions representing the operation of the electronic circuit over time.

6. A simulation method for simulating an operation of an electronic circuit, comprising the steps of:

a step of selecting a data processing means having the highest processing speed and the lowest convergence capacity, of a plurality of data processing means for executing a simulation of an electronic circuit operation, in various method different in processing speed and convergence capacity of a solution, by use of an implicit integration formula and a Newton iteration method, thereby performing a simulation;

a step of detecting iteration times by the Newton iteration method in the processing of said data processing means which performed the simulation; and

a step of controlling said processing selecting step so as to sequentially switch said data processing means to that one having the higher processing speed and the lower convergence capacity next to said data processing means which performed the simulation, when the iteration times detected by said time-detecting step is beyond a predetermined allowed time;

wherein one of the processing by said plurality of data processing means further includes

a step of receiving description data of an electronic circuit to be simulated,

a step of partitioning the description data of the electronic circuit received from said data input step to generate description data of partial circuits,

a step of deciding a time point of the next stage for the description data of the partial circuits generated by said data partitioning step,

a step of generating a simultaneous linear equation by application of the implicit integration formula and the Newton iteration method to the description data of the partial circuits at the time point of the next stage as decided by said point deciding step,

a step of performing incomplete LU factorization processing in parallel on each of

several blocks in a bordered-block-diagonal coefficient matrix of the simultaneous linear equation generated by said equation generating step to produce a plurality of fill-ins, respectively,

a step of respectively adding a fill-in generated by the corresponding ILU factorization means to a combined portion of coefficient matrices, in parallel,

a step of respectively ILU-factorizing a corresponding one of several line collections on the combined portion, in parallel,

a step of repeating a series of processing by said equation generating step, said block ILU factorization step, said fill-in adding step, and said line collection ILU factorization step until convergence of a solution in the simultaneous linear equation generated by said equation generating step is reached,

a step of repeating a series of iteration processing by said equation generating step, said block ILU factorization step, said fill-in adding step, said line collection ILU factorization step, an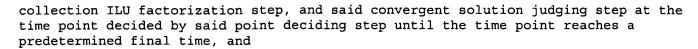d said convergent solution judging step at the time point decided by said point deciding step until the time point reaches a predetermined final time, and

a step of outputting a series of convergent solutions representing the operation of the electronic circuit over time.

7. A computer readable memory storing a control program for controlling a simulation device for simulating an operation of an electronic circuit, the control program comprising the steps of:

a step of receiving description data of an electronic circuit to be simulated;

a step of partitioning the description data of the electronic circuit received from said data input step to generate description data of partial circuits;

a step of deciding a time point of the next stage for the description data of the partial circuits generated by said data partitioning step;

a step of generating a simultaneous linear equation by application of an implicit integration formula and a Newton iteration method to the description data of the partial circuits at the time point of the next stage as decided by said point deciding step;

a step of performing incomplete LU factorization processing in parallel on each of several blocks in a bordered-block-diagonal coefficient matrix of the simultaneous linear equation generated by said equation generating step to produce a plurality of fill-ins, respectively;

a step of respectively adding a fill-in generated by the corresponding ILU factorization means to a combined portion of coefficient matrices, in parallel;

a step of respectively ILU-factorizing a corresponding one of several line collections on the combined portion, in parallel;

a step of repeating a series of processing by said equation generating step, said block ILU factorization step, said fill-in adding step, and said line collection ILU factorization step until convergence of a solution in the simultaneous linear equation generated by said equation generating step is reached;

a step of repeating a series of iteration processing by said equation generating step, said block ILU factorization step, said fill-in adding step, said line collection ILU factorization step, and said convergent solution judging step at the

time point decided by said point deciding step until the time point reaches a predetermined final time; and

a step of outputting a series of convergent solutions representing the operation of the electronic circuit over time.

8. A computer readable memory storing a control program for controlling a simulation device for simulating an operation of an electronic circuit, the control program comprising the steps of:

a step of selecting a data processing means having the highest processing speed and the lowest convergence capacity, of a plurality of data processing means for executing a simulation of an electronic circuit operation, in various methods different in processing speed and convergence capacity of a solution, by use of an implicit integration formula and a Newton iteration method, thereby performing a simulation;

a step of detecting iteration times by the Newton iteration method in the processing of said data processing means which performed the simulation; and

a step of controlling said processing selecting step so as to sequentially switch said data processing means to that one having the higher processing speed and the lower convergence capacity next to said data processing means which performed the simulation, when the iteration times detected by said time-detecting step is beyond a predetermined allowed time;

wherein one of the processing by said plurality of data processing means further includes

a step of receiving description data of an electronic circuit to be simulated,

a step of partitioning the description data of the electronic circuit received from said data input step to generate description data of partial circuits,

a step of deciding a time point of the next stage for the description data of the partial circuits generated by said data partitioning step,

a step of generating a simultaneous linear equation by application of the implicit integration formula and the Newton iteration method to the description data of the partial circuits at the time point of the next stage as decided by said point deciding step,

a step of performing incomplete LU factorization processing in parallel on each of several blocks in a bordered-block-diagonal coefficient matrix of the simultaneous linear equation generated by said equation generating step to produce a plurality of fill-ins, respectively,

a step of respectively adding a fill-in generated by the corresponding ILU factorization means to a combined portion of coefficient matrices, in parallel,

a step of respectively ILU-factorizing a corresponding one of several line collections on the combined portion, in parallel,

a step of repeating a series of processing by said equation generating step, said block ILU factorization step, said fill-in adding step, and said line collection ILU factorization step until convergence of a solution in the simultaneous linear equation generated by said equation generating step is reached

a step of repeating a series of iteration processing by said equation generating step, said block ILU factorization step, said fill-in adding step, said line

collection ILU factorization step, and said convergent solution judging step at the time point decided by said point deciding step until the time point reaches a predetermined final time, and

a step of outputting a series of convergent solutions representing the operation of the electronic circuit over time.

☐ Generate Collection   Print

L3: Entry 4 of 8   File: USPT   Jan 30, 2001

US-PAT-NO: 6182270
DOCUMENT-IDENTIFIER: US 6182270 B1

TITLE: Low-displacement rank preconditioners for simplified non-linear analysis of circuits and other devices

DATE-ISSUED: January 30, 2001

INVENTOR-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY
Feldmann; Peter	Short Hills	NJ		
Long; David Esley	Chatham	NJ		
Melville; Robert C.	New Providence	NJ		

ASSIGNEE-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY	TYPE CODE
Lucent Technologies Inc.	Murray Hill	NJ			02

APPL-NO: 08/ 975250   [PALM]
DATE FILED: November 20, 1997

PARENT-CASE:
RELATED APPLICATION The present application claims the benefit of U.S. Provisional Application No. 60/032,388, filed Dec. 4, 1996 and entitled "Low-Displacement Rank Preconditioners for Periodic and Quasi-Periodic Boundary Value Problems."

INT-CL: [07] G06 F 17/50, G06 F 17/11, G06 F 17/16

US-CL-ISSUED: 716/5; 703/2, 703/14, 708/802
US-CL-CURRENT: 716/5; 703/14, 703/2, 708/802

FIELD-OF-SEARCH: 395/500.02-500.19, 395/500.23, 395/500.34-500.37, 708/801-804, 716/1-21, 703/13-16, 703/2

PRIOR-ART-DISCLOSED:

U.S. PATENT DOCUMENTS

Search Selected   Search ALL   Clear

	PAT-NO	ISSUE-DATE	PATENTEE-NAME	US-CL
☐	5136538	August 1992	Karmarkar et al.	708/607
☐	5604911	February 1997	Ushiro	703/2

☐ <u>5867416</u>   February 1999   Feldmann et al.   708/801

OTHER PUBLICATIONS

M. Rosch, "Schnelle des Stationare Simulation Nichtlinearer Schaltungen," Dissertation, Technical University of Munich, 1992, vol. 46, No. 3, May 1992, pp. 168-176.

PTO 99-5358 (English translation of the article by M. Rosch and K. Antreich, "Schnelle Stationare Simulation Nichtlinearer Schaltungen im Freuenzebereich," AEU, vol. 46, No. 3, pp. 1680176, 1992, translated by The Ralph McElroy Translation Company, Sep. 1999.

PTO 99-5369 (English translation of the German article by M. Rosch, "Schnelle Simulation des Stationaren Verhaltens Nichtlinearer Schaltungen," Dissertation, Technical University of Munich, 1992, translated by Schreiber Translations, Inc., Translated Nov. 1999.

Feldmann et al. ("Efficient Frequency Domain Analysis of Large Nonlinear Analog Circuits", Proceedings of the IEEE Custom Integrated Circuits Conference, May 1996, pp. 1-4).

Brachtendorf et al. ("A simulation tool for the analysis and verification of the steady state of circuit designs", International Journal of Circuit Theory and Applications, vol. 23, pp. 311-323, Jan. 1995).

Gremban et al. ("Performance Evaluation of a New Parallel Preconditioner", NTIS, 34 pages, Oct. 1, 1994).

Gover et al. ("Displacement rank and quasitriangular decomposition for r-Toeplitz matrices", Linear Algebra Applic., vol. 88-89, pp. 329-348, Jan. 1, 1987.

K.S. Kundert, J.K. White and A. Sangiovanni-Vincentelli, "Steady-State Methods for Simulating Analog and Microwave Circuits," Kluwer, Boston, MA, pp. 91-93, 124-132, 1990.

V. Rizzoli, C. Cecchetti, A. Lipparini and F. Mastri, "General-Purpose Harmonic Balance Analysis of Nonlinear Microwave Circuits Under Multitone Excitation," IEEE Trans. Microwave Theory and Tech., vol. MTT-36, No. 12, pp. 1650-1660, Dec. 1988.

V. Rizzoli, F. Mastri, F. Sgallari and V. Frontini, "The Exploitation of Sparse Matrix Techniques in Conjunction with the Piece- Wise Harmonic Balance Method for Nonlinear Microwave Circuit Analysis," IEEE MTT-S Int. Microwave Symp. Digest, Dallas, TX, pp. 1295-1298, May 1990.

D. Long, R. Melville, K. Ashby and B. Horton, "Full-chip Harmonic Balance," Proceedings of the IEEE Custom Integrated Circuits Conference, May 1997.

P. Feldmann, R. Melville and D. Long, "Efficient Frequency Domain Analysis of Large Non-Linear Analog Circuits," Proceedings of the IEEE Custom Integrated Circuits Conference, May 1996.

R. Melville, P. Feldmann and J. Roychowdhury, "Efficient Multi-Tone Distortion Analysis of Analog Integrated Circuits,"Proceedings of the IEEE Custom Integrated Circuit Conference, May 1995, pp. 1-4.

M. Rosch and K. Antreich, "Schnelle Stationare Simulation Nichtlinearer Schaltungen im Freuenzebereich," AEU, vol. 46, No. 3, pp. 168-176, 1992.

T. Kailath and A. Sayed, "Displacement Structure: Theory and Applications," SIAM Review, vol. 37, No. 3, pp. 297-386, Sep. 1995.

I. Gohberg and V. Olshevsky, "Circulants, Displacements and Decompositions of Matrices," Integr. Equat. Oper. Th, vol. 15, pp. 730-743, Birkhauser Verlag, Basel, 1992.

I. Gohberg and V. Olshevsky, "Complexity of Multiplication With Vectors for Structures Matrices," Linear Algebra and its Applications, 202, pp. 163-192, 1994.

B. Troyanovsky, Z. Yu, L. So and R.W. Dutton, "Relaxation-Based Harmonic Balance Technique for Semiconductor Device Simulation," Proceedings of the IEEE/ACM International Conference on Computer-Aided Design, pp. 700-703, Nov. 1995.

H.G. Brachtendorf, G. Welsch, R. Laur and A. Buse-Gerstner, "Numerical Steady State Analysis of Electronic Circuits Driven by Multi-Tone Signals," Electrical Engineering 79, pp. 103-112, Springer-Verlag, 1996.

H.G. Brachtendorf, G. Welsch, and R. Laur, "A Simulation Tool for the Analysis and

Verification of the Steady State of Circuit Designs," International Journal of Circuit Theory and Applications, vol. 23, pp. 311-323, 1995.
K.S. Kundert and A. Sangiovanni-Vincentelli, "Simulation on Nonlinear Circuits in the Frequency Domain," IEEE Transactions on Computer-Aided Design, vol. CAD-5, No. 4, pp. 521-535, Oct. 1986.
V. Rizzoli, A. Lipparini, A. Costanzo, F. Mastri, C. Cecchetti, A. Neri and D. Masotti, "State of the Art Harmonic Balance Simulation of Forced Nonlinear Microwave Circuits by the Piecewise Technique," IEEE Trans. Microwave Theory and Tech., vol. 40, No. 1, pp. 12-28, Jan. 1992.
V. Rizzoli, A. Costanzo, P.R. Ghigi, F. Mastri, D. Masotti and C. Cecchetti, "Recent Advances in Harmonic-Balance Techniques for Nonlinear Microwave Circuit Simulation,"AEU, vol. 46, No. 4, pp. 286-297, 1992.

ART-UNIT: 278

PRIMARY-EXAMINER: Lintz; Paul R.

ASSISTANT-EXAMINER: Kik; Phallaka

ATTY-AGENT-FIRM: Ryan, Mason & Lewis, LLP

ABSTRACT:

Methods and apparatus for performing non-linear analysis using preconditioners to reduce the computation and storage requirements associated with processing a system of equations. A circuit, system or other device to be analyzed includes n unknown waveforms, each characterized by N coefficients in the system of equations. A Jacobian matrix representative of the system of equations is generated. The Jacobian matrix may be in the form of an n.times.n sparse matrix of dense N.times.N blocks, such that each block is of size N.sup.2. In an illustrative embodiment, a low displacement rank preconditioner is applied to the Jacobian matrix in order to provide a preconditioned linear system. The preconditioner may be in the form of an n.times.n sparse matrix which includes compressed blocks which can be represented by substantially less than N.sup.2 elements. For example, the compressed blocks may each be in the form of a low displacement rank matrix corresponding to a product of two generator matrices having dimension N.times..alpha., where .alpha.<<N. The preconditioned linear system may be solved by factoring the preconditioner using a sparse lower-upper (LU) factorization or other similar sparse factorization method applied to the compressed blocks.

23 Claims, 8 Drawing figures

☐   **Generate Collection**   **Print**

L3: Entry 4 of 8   File: USPT   Jan 30, 2001

DOCUMENT-IDENTIFIER: US 6182270 B1
TITLE: Low-displacement rank preconditioners for simplified non-linear analysis of circuits and other devices

Detailed Description Text (31):
After the symmetric permutation, a standard sparse lower-upper (LU) factorization is performed on SJS.sup.-1, treating the N.times.N blocks as arithmetic "elements." FIG. 3 shows the symbolic structure of a Jacobian matrix J in a relatively simple one-tone harmonic balance example in which n=13 and N=63. The elements which are pure diagonals, as indicated by the diagonal lines in FIG. 3, are g entries without a corresponding c entry. This is consistent with the above-noted statement that there are generally more symbolic g entries than c entries. Block fill-in generated during the factorization will generally be the same as that encountered when doing a scalar factorization on a matrix with the same symbolic structure. In typical circuit analysis applications, the block fill-in involves computation which may be on the order of about 5n or so. However, it should be noted that dense manipulation of the N.times.N elements can cause the computation time to increase dramatically to O(nN.sup.3) with a storage space requirement of O(nN.sup.2). The use of low displacement rank preconditioners in accordance with the invention ensures that the element arithmetic can be performed without dense manipulation and therefore with reasonable computation time and memory requirements.

Detailed Description Text (52):
Using the FFT, a matrix-vector product with an N-dimensional factor circulant can be accomplished in time O(NlogN) rather than O(N.sup.2), which allows an apply of Equation (19) in time O(.alpha.NlogN). Moreover, the inverse of J.sub..beta. has substantially the same form with a series representation of the same length. The series length .alpha. is within about a factor of two of .beta., the number of averaging sections. The inversion of the preconditioned matrix J.sub..beta. may be accomplished with an object-oriented extension of a conventional sparse LU factorization which manipulates arithmetic "elements" rather than floating point numbers. Each element of the factorization may be stored in the series of Equation (19). The inversion time and apply time for the inverted preconditioned matrix J.sub..beta..sup.-1 are roughly .beta. times more expensive than the time to invert or apply the block diagonal preconditioner, which may be viewed as the frequency domain form of J.sub.1.

☐ ▮ Generate Collection ▮ ▮ Print ▮

L3: Entry 3 of 8   File: USPT   Mar 19, 2002

US-PAT-NO: 6360190
DOCUMENT-IDENTIFIER: US 6360190 B1

TITLE: Semiconductor process device simulation method and storage medium storing simulation program

DATE-ISSUED: March 19, 2002

INVENTOR-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY
Kumashiro; Shigetaka	Tokyo			JP

ASSIGNEE-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY	TYPE CODE
NEC Corporation				JP	03

APPL-NO: 09/ 200061   [PALM]
DATE FILED: November 25, 1998

FOREIGN-APPL-PRIORITY-DATA:

COUNTRY	APPL-NO	APPL-DATE
JP	9-343689	November 28, 1997

INT-CL: [07] G06 F 7/60, G06 F 17/10, G06 F 17/50

US-CL-ISSUED: 703/2; 703/14, 716/4, 716/5, 716/20
US-CL-CURRENT: 703/2; 703/14, 716/20, 716/4, 716/5

FIELD-OF-SEARCH: 703/2, 703/14, 716/20, 716/4-5

PRIOR-ART-DISCLOSED:

U.S. PATENT DOCUMENTS

▮ Search Selected ▮ ▮ Search ALL ▮ ▮ Clear ▮

	PAT-NO	ISSUE-DATE	PATENTEE-NAME	US-CL
☐	5148379	September 1992	Konno et al.	716/20
☐	5490278	February 1996	Mochizuki	708/446
☐	5557710	September 1996	Amdursky et al.	345/419
☐	5604911	February 1997	Ushiro	703/2
☐	5684723	November 1997	Nakadai	716/20

☐	<u>5844821</u>	December 1998	Roychowdhury	716/4
☐	<u>6144932</u>	November 2000	Hachiya	703/14
☐	<u>6182270</u>	January 2001	Feldmann et al.	716/5

FOREIGN PATENT DOCUMENTS

FOREIGN-PAT-NO	PUBN-DATE	COUNTRY	US-CL
KOKAI 6-53155	February 1994	JP	
KOKAI 8-213334	August 1996	JP	

OTHER PUBLICATIONS

Traar et al.: "High performance preconditioning on supercomputers for the 3D device simulator MINIMOS"; IEEE Proc. Supercomputer '90; pp. 224-231, Nov. 1990.*
Zhao et al.: "A new preconditioner for CGS iteration in solving large sparse nonsymmetric linear equations in semiconductor device simulation"; IEEE Trans. Comp. Aided Design of IC and Systems; pp. 1432-1440, Nov. 1991.*
Dag et al.; "Toward improved uses of the conjugate gradient method for power system applications"; IEEE Trans. Power Systems; pp. 1306-1314; Aug. 1997.*
Eickhoff et al.; "Levelized incomplete LU factorization and its application to large-scale circuit simulation"; IEEE Trans. Computer-aided Design of Integrated Circuits and Systems; pp. 720-727; Jun. 1995.*
Fischer et al.; "Optimum scaling of non-symmetric Jacobian matrices for threshold pivoting preconditioners"; NUPAD V. Int. Workshop on Numerical Modeling of Processes and Devices for Integrated Circuits; pp. 123-126; Jun. 1994.*
Burch et al.; "A new matrix solution technique for general circuit simulation"; IEEE Trans. Computer-aided Design of Integrated Circuits and Systems; pp. 225-241; Feb. 1993.*
Pakzad et al.; "Independent columns: a new parallel ILU preconditioner for the PCG method"; Parallel Computing; vol. 23; pp. 637-647; Jun. 1997.*
Chronopoulos et al.; Parallel solution of a traffic flow simulation problem; Parallel Computing; vol. 22; pp. 1965-1983; Mar. 1997.*
"Process Device Simulation Technology", by Ryo Dan, Apr. 20, 1990,pp. 105-134.
"A New High-Speed Non-equilibrium Point Defect Model for Annealing Simulation" by M.Kawakami Et Al.
"Rapid Convergence Method for Bipolar-MOS Composit Device Simulator TONADDE II" by S.Nakamura et al.

ART-UNIT: 2123

PRIMARY-EXAMINER: Teska; Kevin J.

ASSISTANT-EXAMINER: Jones; Hugh

ATTY-AGENT-FIRM: Hayes, Soloway, Hennessey, Grossman & Hage, P.C.

ABSTRACT:

In this semiconductor process device simulation method, a coefficient <u>matrix</u> constituted by a principal <u>diagonal</u> submatrix arranged at any one of principal diagonals corresponding to each mesh point and representing a self feedback function at the mesh point, the principal <u>diagonal</u> submatrix having rows and columns in numbers corresponding to the number of mesh points, and a non-principal <u>diagonal</u> submatrix arranged on any one of a row and column passing through principal <u>diagonal</u> positions corresponding to the mesh point and representing an

interaction between the mesh point corresponding to the principal diagonal positions and an adjacent mesh point connected to the mesh point through a mesh branch is generated. Calculation for the submatrices is performed while regarding each submatrix of the coefficient matrix as one element, thereby performing incomplete LU factorization of the coefficient matrix.

2 Claims, 9 Drawing figures

☐ ▮ Generate Collection ▮   ▮ Print ▮

L3: Entry 3 of 8   File: USPT   Mar 19, 2002

DOCUMENT-IDENTIFIER: US 6360190 B1
TITLE: Semiconductor process device simulation method and storage medium storing
simulation program

Abstract Text (1):
In this semiconductor process device simulation method, a coefficient matrix
constituted by a principal diagonal submatrix arranged at any one of principal
diagonals corresponding to each mesh point and representing a self feedback
function at the mesh point, the principal diagonal submatrix having rows and
columns in numbers corresponding to the number of mesh points, and a non-principal
diagonal submatrix arranged on any one of a row and column passing through
principal diagonal positions corresponding to the mesh point and representing an
interaction between the mesh point corresponding to the principal diagonal
positions and an adjacent mesh point connected to the mesh point through a mesh
branch is generated. Calculation for the submatrices is performed while regarding
each submatrix of the coefficient matrix as one element, thereby performing
incomplete LU factorization of the coefficient matrix.

Brief Summary Text (20):
This is because in making the computer calculations, a square submatrix having a
uniform size of n.times.n is used as the processing unit of incomplete LU-
factorization. For example, in the above-described simulation, the submatrix size
of n.times.n is maintained even at a mesh point where n equations are not defined.
The computer performing the calculations must form an overall coefficient matrix
while inserting "1" to the corresponding principal diagonal portions of the
submatrix and set the coefficient matrix on the memory. For this reason, excess
memory capacity is used.

Brief Summary Text (25):
In order to achieve the above objects, according to one aspect of the present
invention there is provided a semiconductor device manufacturing process simulation
method to aid manufacturers in forecasting electrical characteristics of
semiconductor devices by performing a plurality of matrix manipulations of terms
representing physical properties of the semiconductor devices, the matrices to be
manipulated representing multidimensional simultaneous linear equations that are to
be solved by a matrix solver that uses an iterative method in which the matrices
are preconditioned by incomplete LU-factorization, the method for generating fill-
ins for the matrices comprising the steps of: a first step of dividing the surface
of a semiconductor device to be processed into a plurality of rectangles forming a
mesh of predetermined size; a second step of assigning a numerical value to each
mesh point of the mesh; a third step of setting equations representing a
relationship among the plurality of numerical values; a fourth step of generating a
coefficient matrix constituted by a plurality of principal diagonal submatrices
each of which is arranged at each one of principal diagonal positions corresponding
to each mesh point and representing a self feedback function to the mesh point, the
coefficient matrix having rows and columns in numbers corresponding to the number
of mesh points, and a plurality of non-principal diagonal submatrices each of which
is arranged on any one of rows and columns passing through the principal diagonal
positions corresponding to the mesh point and representing an interaction between
the mesh point corresponding to the principal diagonal located on the same row or

●   ●

on the same column of the coefficient matrix with the non-principal diagonal submatrix and an adjacent mesh point connected to the mesh point through a mesh branch; a fifth step of performing calculation for the submatrices while regarding each submatrix of the coefficient matrix as one element, thereby performing incomplete LU-factorization of the coefficient matrix, wherein each of the principal diagonal submatrices is a square having rows and columns equal in number to equations set for a mesh point corresponding to the principal diagonal submatrix, each of the non-principal diagonal submatrices arranged in a row direction in correspondence with each of the mesh points is a matrix having rows equal in number to equations set at a mesh point corresponding to the principal diagonal submatrix located in the row and columns equal in number to equations set at an adjacent mesh point connected to the mesh point through a mesh branch, and each of the non-principal diagonal submatrices arranged in a column direction in correspondence with each of the mesh points is a matrix having columns equal in number to equations set at a mesh point corresponding to the principal diagonal submatrix located in the column and rows equal in number to the equations set at an adjacent mesh point connected to the mesh point through a mesh branch; and a sixth step of producing a signal indicative of the result of said calculation.

Brief Summary Text (26):
In another aspect of the invention there is provided a computer readable memory storing a semiconductor device manufacturing process simulation program to aid manufacturers in forecasting electrical characteristics of semiconductor devices by performing a plurality of matrix manipulations of terms representing physical properties of the semiconductor devices, the matrices to be manipulated representing multidimensional simultaneous linear equations that are to be solved by a matrix solver that uses an iterative method in which the matrices are preconditioned by incomplete LU-factorization, the program including a routine for generating fill-ins for the matrices, the routine comprising the steps of: causing a computer to perform the following functions: a first function of dividing a surface of a semiconductor device to be processed into a mesh of predetermined size; a second function of assigning a numerical value to each mesh point of the mesh; a third function of setting equations representing a relationship among the plurality of numerical values; a fourth function of generating a coefficient matrix constituted by a plurality of principal diagonal submatrices each of which is arranged at each one of principal diagonal positions corresponding to each mesh point and representing a self feedback function at the mesh point, the coefficient matrix having rows and columns in numbers corresponding to the number of mesh points, and a plurality of non-principal diagonal submatrices each of which is arranged on any one of rows and columns and representing an interaction between the mesh point corresponding to the principal diagonal positions located on the same row or on the same column of the coefficient matrix with the non-principle diagonal submatrix and an adjacent mesh point connected to the mesh point through a mesh branch; and a fifth function of performing calculation for the submatrices while regarding each submatrix of the coefficient matrix as one element; thereby performing incomplete LU-factorization of the coefficient matrix, wherein each of the principal diagonal submatrices is a square matrix having rows and columns equal in number to equations set forth for a mesh point corresponding to the principal diagonal submatrix, each of the non-principal diagonal submatrices arranged in a row direction in correspondence with each of the mesh points is a matrix having rows equal in number to equations set at a mesh point corresponding to the principal diagonal submatrix and located in the row and columns equal in number to equations set at an adjacent mesh point connected to the mesh point through a mesh branch, and each of the non-principal diagonal submatrices arranged in a column direction in correspondence with each of the mesh points is a matrix having columns equal in number to equations set a mesh point corresponding tot he principal diagonal submatrix located in the column and rows equal in number to equations set at an adjacent mesh point connected to the mesh point through a mesh branch.

Detailed Description Text (59):

According to the present invention, there is provided the semiconductor process device simulation method comprising a first step of dividing the surface of a semiconductor device to be processed into a mesh of predetermined size, a second step of assigning a numerical value to each mesh point of the mesh, a third step of setting equations representing a relationship among the plurality of numerical values, a fourth step of generating a coefficient matrix constituted by a principal diagonal submatrix arranged at any one of principal diagonals corresponding to each mesh point and representing a self feedback function at the mesh point, the principal diagonal submatrix having rows and columns in numbers corresponding to the number of mesh points, and a non-principal diagonal submatrix arranged on any one of a row and column passing through principal diagonal positions corresponding to the mesh point and representing an interaction between the mesh point corresponding to the principal diagonal positions and an adjacent mesh point connected to the mesh point through a mesh branch, and a fifth step of performing calculation for the submatrices while regarding each submatrix of the coefficient matrix as one element, thereby performing incomplete LU-factorization of the coefficient matrix, wherein the principal diagonal submatrix is a square matrix having rows and columns equal in number to equations set for a mesh point corresponding to the principal diagonal submatrix, the non-principal diagonal submatrix being arranged in a row direction in correspondence with each of the mesh points is a matrix having rows equal in number to equations set at a mesh point corresponding to the principal diagonal submatrix and columns equal in number to equations set at an adjacent mesh point connected to the mesh point through a mesh branch, and the non-principal diagonal submatrix being arranged in a column direction in correspondence with each of the mesh points is a matrix having columns equal in number to equations set at a mesh point corresponding to the principal diagonal submatrix and rows equal in number to equations set at an adjacent mesh point connected to the mesh point through a mesh branch.

End of Result Set

☐ ▐ Generate Collection ▌ ▐ Print ▌

L7: Entry 5 of 5   File: USPT   Feb 6, 1996

US-PAT-NO: 5490278
DOCUMENT-IDENTIFIER: US 5490278 A

TITLE: Data processing method and apparatus employing parallel processing for
solving systems of linear equations

DATE-ISSUED: February 6, 1996

INVENTOR-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY
Mochizuki; Yoshiyuki	Osaka			JP

ASSIGNEE-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY	TYPE CODE
Matsushita Electric Industrial Co., Ltd.	Osaka			JP	03

APPL-NO: 07/ 912180   [PALM]
DATE FILED: July 13, 1992

FOREIGN-APPL-PRIORITY-DATA:

COUNTRY	APPL-NO	APPL-DATE
JP	3-172168	July 12, 1991
JP	3-173616	July 15, 1991

INT-CL: [06] G06 F 17/16, G06 F 17/11

US-CL-ISSUED: 395/800; 364/735, 364/931.4, 364/931.41, 364/937.1, 364/DIG.2
US-CL-CURRENT: 708/446

FIELD-OF-SEARCH: 395/800, 364/735, 364/754, 364/724.05, 364/807

PRIOR-ART-DISCLOSED:

U.S. PATENT DOCUMENTS

▐ Search Selected ▌ ▐ Search ALL ▌ ▐ Clear ▌

	PAT-NO	ISSUE-DATE	PATENTEE-NAME	US-CL
☐	4787057	November 1988	Hammond	364/754
☐	5113523	May 1992	Colley et al.	395/800
☐	5274832	December 1993	Khan	395/800

☐ 5301342   April 1994   Scott   395/800

OTHER PUBLICATIONS

B. Smith et al, "Sparse Matrix Computations on an FFP Machine," IEEE, Oct. 1988, pp. 215-218.

A. El-Amawy et al, "Efficient Linear and Bilinear Arrays for Matrix Triangularisation with Partial Pivoting," IEE Proceedings, vol. 137, No. 4, Jul. 1990, pp. 295-300.

Transaction Of The Institute Of Electronics, Information And Communications Engineers Of Japan, vol. 72, No. 12, Dec. 1989, Tokyo, Japan, pp. 1336-1343, Nobuyuki Tanaka et al. "Special Parallel Machine for LU Decomposition of a Large Scale Circuit Matrix and Its Performance".

Parallel Computing, vol. 11, No. 4, Aug. 1989, Amsterdam, NL, pp. 201-221, Gita Alaghband, "Parallel Pivoting Combined with Parallel Reduction and Fill-In Control".

IEEE Transactions On Computers, vol. 32, No. 12, Dec. 1983, New York, US, pp. 1109-1117, Mandayam A. Srinivas, "Optimal Parallel Scheduling of Gaussian Elimination DEG's".

Parallel Computing, vol. 13, No. 3, Mar. 1990, Amsterdam, NL, pp. 289-294, Hyoung Joong Kim et al., "A Parallel Algorithm Solving Tridiagonal Toeplitz Linear System".

Proceedings Of The 1989 Power Industry Computer Application Conference, IEEE Press, New York, US, 1 May 1989, Seattle, Washington, US, pp. 9-15, D. C. Yu et al., "A New Approach for the Forward and Backward Substitution of Parallel Solution of Sparse Linear Equations Based on Dataflow Architecture".

Proceedings Of The IEEE 1983 International Symposium On Circuits And Systems, IEEE Press, New York, US, vol. 1/3, 2 May 1983, Newport Beach, California, US, pp. 214-217, R. M. Kieckhafer et al., "A Clustered Processor Array for the Solution of the Unstructured Sparse Matrix Equations".

Proceedings Of The 1986 IBM Europe Instutute Seminar On Parallel Computing, North Holland Amsterdam NL, 11 Aug. 1986, Oberlech, Austria, pp. 99-106, Iain S. Duff, "Parallelism in Sparse Matrices".

ART-UNIT: 237

PRIMARY-EXAMINER: Black; Thomas G.

ASSISTANT-EXAMINER: Harrity; Paul

ATTY-AGENT-FIRM: Wenderoth, Lind & Ponack

ABSTRACT:

A linear calculating equipment comprises a memory for storing a coefficient matrix, a known vector and an unknown vector of a given system of linear equations, a pivoting device for choosing pivots of the matrix, a plurality of preprocessors for executing K steps of preprocessing for multi-pivot simultaneous elimination, an updating device for updating the elements of the matrix and the components of the vectors, a register set for storing values of the variables, a back-substitution device for obtaining a solution and a main controller for controlling the linear calculating equipment as a whole.

13 Claims, 23 Drawing figures

End of Result Set

☐ ▮▮▮ Generate Collection ▮▮▮ ▮ Print ▮

L7: Entry 5 of 5   File: USPT   Feb 6, 1996

DOCUMENT-IDENTIFIER: US 5490278 A
TITLE: Data processing method and apparatus employing parallel processing for
solving systems of linear equations

Brief Summary Text (8):
A similar algorithm to multi-pivot simultaneous elimination algorithms is described
in Jim Armstrong, "Algorithm and Performance Notes for Block LU Factorization,"
International Conference on Parallel Processing, 1988, Vol. 3, pp 161-164. It is a
block LU factorization algorithm intended to speed up matrix operations and should
be implemented in vector computers or computers with a few multiplexed processors.

Brief Summary Text (98):
According to another aspect of the present invention, there is provided a parallel
elimination method for solving the system of linear equations (2) in a parallel
computer comprising C clusters CL.sub.1, . . . , CL.sub.C connected by a network.
Each of the clusters comprises P.sub.c element processors and a shared memory that
stores part of the reduced matrices A.sup.(r) and the known vectors b.sup.(r) and
the unknown vector x. The method comprises:

Brief Summary Text (99):
a data distribution means that distributes the rows of the coefficient matrix
A.sup.(0) and the components of b.sup.(0) and x to the shared memory of the
clusters in such a manner as each block of consecutive k rows and corresponding 2k
components is transmitted to the shared memory in the cyclic order of CL.sub.1, . .
. , CL.sub.C, CL.sub.1, CL.sub.2, . . . , and assigns those distributed to the
cluster's shared memory to its element processors row by row,

Brief Summary Text (103):
in the element processor in charge of the (kP.sub.c +1)th row, transmits the
results to the shared memory of every other cluster to which the element processor
in charge of an i-throw such that kP.sub.c +1.ltoreq.i.ltoreq.n belongs, and, for
l=2, . . . , P.sub.c, calculates ##EQU4## for kP.sub.c +1.ltoreq.i.ltoreq.n in the
element processor in charge of the i-th row, calculates ##EQU5## in the element
processor in charge of the (kP.sub.c +1)th row, and, after the pivot choosing means
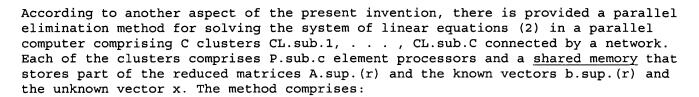determines the pivot

Brief Summary Text (105):
in the element processor in charge of the (kP.sub.c +1)th row, transmits the
results (38) and (39) to the shared memory of every other cluster to which the
element processor in charge of an i-th row such that kP.sub.c
+1+1.ltoreq.i.ltoreq.n belongs,

Brief Summary Text (113):
an elementary back-transmission means that transmits x.sub.i to the shared memory
of every cluster to which the element processor in charge of an h-th row such that
1.ltoreq.h.ltoreq.i-1 belongs,

Brief Summary Text (119):

According to another aspect of the present invention, there is provided a parallel elimination method for solving the system of linear equations (2) in a parallel computer comprising C clusters CL.sub.1, . . . , CL.sub.C connected by a network. Each of the clusters comprises P.sub.c element processors and a <u>shared memory</u> that stores part of the reduced matrices A.sup.(r) and the known vectors b.sup.(r) and the unknown vector x. The method comprises:

<u>Brief Summary Text</u> (120):
a data distribution means that distributes the rows of the coefficient matrix A.sup.(0) and the components of b.sup.(0) and x to the clusters in such a manner as each block of consecutive k rows and corresponding 2k components is transmitted to the <u>shared memory</u> in the cyclic order of CL.sub.1, . . . , CL.sub.C, CL.sub.1, CL.sub.2, . . . , and assigns those distributed to the cluster's <u>shared memory</u> to its element processors row by row,

<u>Brief Summary Text</u> (122):
an elementary pre-elimination means that, after the pivot choosing means chooses the pivot (31), calculates (32) and (33) in the element processor in charge of the (P.sub.c k+1)th row, transmits the results to the <u>shared memory</u> of every other cluster to which the element processor in charge of an i-th row such that kP.sub.c +2.ltoreq.i.ltoreq.n belongs, and, for l=2, . . . , P.sub.c, calculates (34) for kP.sub.c +1.ltoreq.i.ltoreq.n in the element processor in charge of the i-th row, calculates (35) and (36) in the element processor in charge of the (kP.sub.c +1)th row, and, after the pivot choosing means chooses the pivot (37), calculates (38) and (39) in the element processor in charge of the (kP.sub.c +1)th row, and transmits the results (38) and (39) to the <u>shared memory</u> of every other cluster to which an element processor in charge of the i-th row such that kP.sub.c +1+1.ltoreq.i.ltoreq.n belongs, calculates,

<u>Detailed Description Text</u> (67):
FIG. 19 shows a block diagram of an element processor or processor module of a parallel computer that implements the seventh embodiment of the present invention. In FIG. 19, 201 is a gate way; 202 is a cache memory; 203 is a central processing unit; 204 is a local memory; 205 is a shared buss. FIG. 20 shows a block diagram of a cluster composed of element processors 212, 213, . . . , 214, a C gateway 210, and a <u>shared memory</u> 211. A network of the parallel computer connects each of the clusters to each other, so that data can be transmitted between any two clusters. Let the number of element processors in each cluster be P.sub.c and the total number of clusters be C. Then the total number P of element processors in the parallel computer is C.multidot.P.sub.c. Furthermore, let the clusters be denoted by CL.sub.1, CL.sub.2, . . . , CL.sub.C, and let the element processors of CL.sub.u be denoted by PR.sub.u 1, . . . , PR.sub.u P.sbsb.c.

<u>Detailed Description Text</u> (75):
In the present first method of pivot choosing, the element processor in charge of each i-th row, by the search means 240, tests if a.sup.(i-1).sub.i i =0. If it is not, then the process terminates. If it is, then the element processor, by the search means 240, searches for a nonzero element in the i-th row of A.sup.(i-1) from a.sup.(i-1) .sub.i i+1 to a.sup.(i-1).sub.i n in this order. If a.sup.(i-1) .sub.i h is the first such element, then the element processor, by the broadcasting means 241, notifies each element processor of the column number h by a broadcast. Specifically, the element processor either transmits h to a specified word of the <u>shared memory</u> 211 of each cluster, and each element processor refers to the word, or the element processor transmits h to a dedicated bus line, and each element processor fetches h into its local memory 204. Then each element processor, by the element interchange means 242, simultaneously interchanges the element with the column number i with the element with the column number h in the row in its charge. Then two element processors in charge of the i-th component and the h-th component of the unknown vector x respectively interchange these component by the component interchange means 243. The pivot choosing process terminates hereby.

<u>Detailed Description Text</u> (77):
In the process of the elementary pre-elimination means 222, if l=1, then the element processor PR.sub.u 1 in charge of the (kP.sub.c +1)th row in the cluster CL.sub.u, where u=k-[k/C]+1, calculates (32) and (33), and transmits the results to the <u>shared memory</u> of every other cluster to which the element processor in charge of an i-th row such that kP.sub.c +2.ltoreq.i.ltoreq.n belongs. If 2.ltoreq.l.ltoreq.P.sub.c, then each element processor in charge of the i-th row such that kP.sub.c +1.ltoreq.i.ltoreq.n calculates (34), and the element processor PR.sub.u 1 calculates (35) and (36). Then after the pivot choosing means determines the pivot (37), the element processor PR.sub.u 1 calculates (38) and (39) and transmits the results to the <u>shared memory</u> of every other cluster to which the element processor in charge of an i-th row such that kP.sub.c +l+1.ltoreq.i.ltoreq.n belongs.

<u>Detailed Description Text</u> (82):
In the eighth step, an elementary back-transmission means that transmits x.sub.i to the <u>shared memory</u> of every clusters such that the element processor in charge of an h-th row such that 1.ltoreq.h.ltoreq.i-1 belongs.

<u>Detailed Description Text</u> (89):
In the pre-elimination means 232, if l=1, then after the pivot choosing means 221 determines the pivot (31), the element processor PR.sub.u 1 in charge of the (kP.sub.c +1)th row in the cluster CL.sub.u, where u=k-[k/C]+1, calculates (32) and (33), and transmits the results to the <u>shared memory</u> of every other cluster. If 2.ltoreq.l.ltoreq.P.sub.c, then each element processor in charge of the i-th row such that kP.sub.c +1.ltoreq.i.ltoreq.n calculates (34), and the element processor PR.sub.u 1 calculates (35) and (36). Then after the pivot choosing means determines the pivot (37), the element processor PR.sub.u 1 calculates (38) and (39) and transmits the results to the <u>shared memory</u> of every other cluster.

WEST Search History

Hide Items | Restore | Clear | Cancel

DATE: Monday, April 19, 2004

Hide?	Set Name	Query	Hit Count
		DB=PGPB,USPT; PLUR=NO; OP=ADJ	
☐	L11	l1 and vector processor$1	1
☐	L10	l1 and array processor$1	1
☐	L9	l1 and L8	4
☐	L8	parallel computer$1	2307
☐	L7	l1 and L6	5
☐	L6	shared memor$3	10121
☐	L5	diagonal same L4	0
☐	L4	lower upper factorization	3
☐	L3	diagonal same L2	8
☐	L2	matrix same L1	39
☐	L1	lu factorization	53

END OF SEARCH HISTORY

□ | Generate Collection | | Print |

L18: Entry 3 of 6   File: USPT   Feb 17, 2004

US-PAT-NO: 6694343
DOCUMENT-IDENTIFIER: US 6694343 B2

TITLE: Method for solving a large sparse triangular system of linear equations

DATE-ISSUED: February 17, 2004

INVENTOR-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY
Forrest; John Joseph	Peekskill	NY		
Megiddo; Nimrod	Palo Alto	CA		
Tomlin; John Anthony	Sunnyvale	CA		

ASSIGNEE-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY	TYPE CODE
International Business Machines Corporation	Armonk	NY			02

APPL-NO: 09/ 778937   [PALM]
DATE FILED: February 8, 2001

INT-CL: [07] G06 F 7/32, G06 F 7/38

US-CL-ISSUED: 708/520; 708/446
US-CL-CURRENT: 708/520; 708/446

FIELD-OF-SEARCH: 708/520, 708/443-444, 708/446

PRIOR-ART-DISCLOSED:

U.S. PATENT DOCUMENTS

| Search Selected | | Search ALL | | Clear |

	PAT-NO	ISSUE-DATE	PATENTEE-NAME	US-CL
□	4787057	November 1988	Hammond	708/607
□	5200915	April 1993	Hayami et al.	708/520
□	5301342	April 1994	Scott	708/520
□	5905666	May 1999	Hoffman et al.	708/520
□	5983230	November 1999	Gilbert et al.	708/520
□	6446105	September 2002	Washio et al.	708/520

☐ 6470368   October 2002   Garg et al.   708/446

☐ 6601080   July 2003   Garg   708/502

ART-UNIT: 2124

PRIMARY-EXAMINER: Mai; Tan V.

ATTY-AGENT-FIRM: Lacasse & Associates LLC Lacasse; Randy W. McSwain; Marc D.

ABSTRACT:

A computer-based method and system comprising three data structures: partially ordered data structure (or simply ordered data structure), contiguous list v, and vector p, is used for solving a large sparse triangular system of linear equations which utilizes only the non-zero components of a matrix to solve large sparse triangular linear equations and generates explicitly only the non-zero entries of the solution. A list of the row indices of the known non-zero values of x which require further processing is stored in the ordered data structure. Actual non-zero values of x are stored in the contiguous list v and the corresponding pointers to the location of these values are stored in the vector p. The computer-based method manipulates these three matrices to find a solution to an upper or lower sparse triangular system of linear equations. In addition, in the instance a matrix becomes dense (or increases in density) by the presence of many active rows, a partitioning method is described via which the dense matrix problem is solved.

13 Claims, 6 Drawing figures

□   **Generate Collection**   **Print**

L18: Entry 3 of 6   File: USPT   Feb 17, 2004

DOCUMENT-IDENTIFIER: US 6694343 B2
TITLE: Method for solving a large sparse triangular system of linear equations

Brief Summary Text (12):
wherein the elements s.sub.11, s.sub.14, and s.sub.34 are the only non-zero elements in (4). The **diagonal** of a square **matrix** (n.times.n) divides it into two halves and helps define two kinds of sparse matrices: upper-triangular or lower triangular. If the **block** below the **diagonal** consists of zeros, the **matrix** is said to be upper triangular. For example, the **matrix** shown below is upper triangular: ##EQU2##

Brief Summary Text (13):
In contrast to (5), if the **block** above the **diagonal** consists of zeros, then the **matrix** is said to be lower-triangular. An example of a lower triangular **matrix** is shown below: ##EQU3##

Current US Original Classification (1):
708/520

First Hit   Fwd Refs

☐ ▐ Generate Collection ▌ ▐ Print ▌

L22: Entry 2 of 3   File: USPT   Jul 1, 1997

US-PAT-NO: 5644517
DOCUMENT-IDENTIFIER: US 5644517 A

TITLE: Method for performing matrix transposition on a mesh multiprocessor
architecture having multiple processor with concurrent execution of the multiple
processors

DATE-ISSUED: July 1, 1997

INVENTOR-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY
Ho; Ching-Tien	San Jose	CA		

ASSIGNEE-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY	TYPE CODE
International Business Machines Corporation	Armonk	NY			02

APPL-NO: 08/ 496036   [PALM]
DATE FILED: June 28, 1995

PARENT-CASE:
This is a continuation of application Ser. No. 07/965,498 filed on Oct. 22, 1992,
now abandoned.

INT-CL: [06] G06 F 15/173, G06 F 17/16

US-CL-ISSUED: 364/725.02; 395/311, 395/800
US-CL-CURRENT: 708/401; 712/17

FIELD-OF-SEARCH: 364/725, 364/741, 395/311

PRIOR-ART-DISCLOSED:

U.S. PATENT DOCUMENTS

▐ Search Selected ▌   ▐ Search ALL ▌   ▐ Clear ▌

	PAT-NO	ISSUE-DATE	PATENTEE-NAME	US-CL
☐	4769790	September 1988	Yamashita	365/189
☐	4787057	November 1988	Hammond	364/754
☐	4914615	April 1990	Karmarkar et al.	364/754
☐	4918527	April 1990	Penard et al.	358/160

☐ 5101371   March 1992   Iobst   364/736

FOREIGN PATENT DOCUMENTS

FOREIGN-PAT-NO   PUBN-DATE   COUNTRY   US-CL
54-87423   July 1979   JP

OTHER PUBLICATIONS

N. G. Azari, A. W. Bojanczyk and S. Y. Lee, "Synchronous and Asynchronous
Algorithms for Matrix Transposition on MCAP," SPIE vol. 975, Advanced Algorithms &
Architectures for Signal Processing III, pp. 277-288., 1988.
J. O. Eklundh, "A Fast Computer Method for Matrix Transposing," IEEE Transactions
on Computers, pp. 801-803, Jul. 1972.
H. S. Stone, "Parallel Processing with the Perfect Shuffle," IEEE Transactions on
Computers, vol. C-20, No. 2, pp. 153-161, Feb. 1971.
S. L. Johnsson, "Communication Efficient Basic Linear Algebra Computations on
Hypercube Architectures," Journal of Parallel and Distributed Computing 4, pp. 133-
172, 1987.
O. A. McBryan and E. F. Van De Velde, "Hypercube Algorithms and Implementations,"
SIAM J. Sci. Stat. Comput., vol. 8, No. 2, pp. s227-287, Mar. 1987.
Q. F. Stout and B. Wagar, "Passing Messages in Link-Bound Hypercubes," In Hypercube
Multiprocessors, SIAM, 1991.
C. T. Ho and M. T. Raghunath, "Efficient Communication Primitives on Circuit-
Switched Hypercubes," IEEE, 0-8186-2290-3/91/0000/0390, pp. 390-397, 1991.
D. Nassimi and S. Sahni, "An Optimal Routing Algorithm for Mesh-Connected Parallel
Computers," Journal of the Association for Computing Machinery, vol. 27, No. 1, pp.
6-29, Jan. 1980.
S. L. Johnsson and C. T. Ho, "Algorithms for Matrix Transposition on Boolean N-Cube
Configured Ensemble Architectures," SIAM J. Matrix Anal. Appl., vol. 9, No. 3, pp.
419-454, Jul. 1988.
H. Nakano, T. Tsuda, "Optimizing Inter-processor Data Transfers in Transpositions
of Matrices Stored Row-wise on Mesh-connected Parallel Computers," Trans. Inf.
Process. Soc. Jpn. (Japan), vol. 27, No. 3, pp. 348-355, 1986. (Article Published
in Japan).

ART-UNIT: 237

PRIMARY-EXAMINER: Black; Thomas G.

ASSISTANT-EXAMINER: Choules; Jack M.

ATTY-AGENT-FIRM: Pintner; James C. Blair; Philip E.

ABSTRACT:

A matrix transpose method for transposing any size matrix on a 2-dimensional mesh
multi-node system with circuit-switched-like routing in the iterative and recursive
forms. The matrix transpose method involves a two-level decomposition technique of
first partitioning each mesh on a diagonal axis into four submeshes and then
further partitioning each of the four submeshes on the diagonal axis into four
submeshes. The transposition of all off-diagonal submatrices can be performed
concurrently and the transposition of all successive on-diagonal submatrices can be
performed iteratively or recursively.

8 Claims, 10 Drawing figures

<u>First Hit</u>   <u>Fwd Refs</u>

☐   ▨ Generate Collection   Print

L22: Entry 2 of 3   File: USPT   Jul 1, 1997

DOCUMENT-IDENTIFIER: US 5644517 A
TITLE: Method for performing matrix transposition on a mesh multiprocessor architecture having multiple processor with concurrent execution of the multiple processors

<u>Current US Cross Reference Classification</u> (1):
712/17

CLAIMS:

1. A method for transposing a matrix comprising the steps of:

a) performing a <u>block</u> allocation of elements of the <u>matrix</u> to respective nodes of a multi-node computer system having a 2-dimensional N.times.N mesh of interconnected nodes, N being an integer greater than 1, each datablock representing a respective submatrix of the <u>matrix, said matrix having a diagonal</u> axis such that a subset of the nodes is on the <u>diagonal</u> axis;

b) partitioning the mesh into four submeshes, each of size less than or equal to (.left brkt-top.N/2.left brkt-top..times..left brkt-top.N/2.right brkt-top.), such that a first group of the four submeshes is on the diagonal axis of the matrix, and a second group of the four submeshes is off the diagonal axis;

c) further partitioning each submesh of the first group, each submesh being of size N'.times.N', where N' is an integer greater than 1, into four submeshes of size less than or equal to (.left brkt-top.N'/2.right brkt-top..times..left brkt-top.N'/2.right brkt-top.), such that a third group of the submeshes of size .left brkt-top.N'/2.right brkt-top..times..left brkt-top.N'/2.right brkt-top. is on the diagonal axis of the matrix, and a fourth group of the submeshes of size .left brkt-top.N'/2.right brkt-top..times..left brkt-top.N'/2.right brkt-top. is off the diagonal axis of the matrix;

d) concurrently performing a transposition of all datablocks on all of the nodes included on all submeshes of the second and fourth groups; and

e) repeating steps b-e for each of the submeshes in the third group if there exists a submesh of size greater than (1.times.1) on the diagonal axis and the submesh has no partitioned submeshes.

3. A computer program product for use in a multi-node computer system having a 2-dimensional N.times.N mesh of interconnected nodes, N being an integer greater than 1, the computer program product comprising:

a) a recording medium;

b) means, recorded on said medium, for instructing the computer system to

1) perform a <u>block</u> allocation of elements of a <u>matrix</u> to be transposed to respective nodes of the computer system, each datablock representing a respective submatrix of the <u>matrix, said matrix having a diagonal</u> axis such that a subset of

the nodes of the system is on the <u>diagonal</u> axis;

2) partition the mesh into four submeshes, each of size less than or equal to (.left brkt-top.N/2.right brkt-top..times..left brkt-top.N/2.right brkt-top.), such that a first group of the four submeshes is on the diagonal axis of the matrix, and a second group of the four submeshes is off the diagonal axis;

3) further partition each submesh of the first group, each submesh being of size N'.times.N', where N' is an integer greater than 1, into four submeshes of size less than or equal to (.left brkt-top.N'/2.right brkt-top..times..left brkt-top.N'/2.right brkt-top.), such that a third group of the submeshes of size .left brkt-top.N'/2.right brkt-top..times..left brkt-top.N'/2.right brkt-top. is on the diagonal axis of the matrix, and a fourth group of the submeshes of size .left brkt-top.N'/2.right brkt-top..times..left brkt-top.N'/2.right brkt-top. is off the diagonal axis of the matrix;

4) concurrently perform a transposition of all datablocks on all of the nodes included on all submeshes of the second and fourth groups; and

5) repeat the instruction steps 2-5 for each of the submeshes in the third group if there exists a submesh of size greater than (1.times.1) on the diagonal axis and the submesh has no partitioned submeshes.

5. A multi-node computer system having a 2-dimensional N.times.N mesh of interconnected nodes, N being an integer greater than 1, the system comprising:

a) means for performing a <u>block</u> allocation of elements of a <u>matrix</u> to be transposed to the nodes, each datablock representing a respective submatrix of the <u>matrix,</u> <u>said matrix having a diagonal</u> axis such that a subset of the nodes is on the <u>diagonal</u> axis;

b) means for partitioning the mesh into four submeshes, each of size less than or equal to (.left brkt-top.N/2.right brkt-top..times..left brkt-top.N/2.right brkt-top.), such that a first group of the four submeshes is on the diagonal axis of the matrix, and a second group of the four submeshes is off the diagonal axis;

c) means for further partitioning each submesh of the first group, each submesh being of size N'.times.N', where N' is an integer greater than 1, into four submeshes of size less than or equal to .left brkt-top.N'/2.right brkt-top..times..left brkt-top.N'/2.right brkt-top., such that third group of the submeshes of size .left brkt-top.N'/2.right brkt-top..times..left brkt-top.N'/2.right brkt-top. is on the diagonal axis of the matrix, and a fourth group of the submeshes of size .left brkt-top.N'/2.right brkt-top..times..left brkt-top.N'/2.right brkt-top. is off the diagonal axis of the matrix;

d) means for concurrently performing a transposition of all datablocks on all of the nodes included on all submeshes of the second and fourth groups; and

e) means for repeating steps b-e for each of the submeshes in the third group if there exists a submesh of size greater than (1.times.1) on the diagonal axis and the submesh has no partitioned submeshes.

End of Result Set

☐ ▰▰▰ Generate Collection ▰▰▰ ‖ Print ‖

L22: Entry 3 of 3   File: USPT   Oct 20, 1992

US-PAT-NO: 5157778
DOCUMENT-IDENTIFIER: US 5157778 A
** See image for Certificate of Correction **

TITLE: Method and apparatus for circuit simulation using parallel processors
including memory arrangements and matrix decomposition synchronization

DATE-ISSUED: October 20, 1992

INVENTOR-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY
Bischoff; Gabriel P.	Marlboro	MA		
Greenberg; Steven S.	Bolton	MA		

ASSIGNEE-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY	TYPE CODE
Digital Equipment Corporation	Maynard	MA			02

APPL-NO: 07/ 513017   [PALM]
DATE FILED: April 23, 1990

PARENT-CASE:
This is a continuation of co-pending application Ser. No. 06/898,476 filed on Aug.
20, 1986, now abandoned.

INT-CL: [05] G06F 7/38, G06F 9/455, G06F 15/16

US-CL-ISSUED: 395/500; 364/931.41, 364/966.1, 364/937.8, 364/933.8, 364/750.2,
364/DIG.2, 395/800
US-CL-CURRENT: 703/14; 708/603, 712/17

FIELD-OF-SEARCH: 364/2MSFile, 364/9MSFile, 364/750.5, 395/800, 395/500

PRIOR-ART-DISCLOSED:

U.S. PATENT DOCUMENTS

▰▰▰ Search Selected ▰▰▰ ‖ ▰▰▰ Search ALL ▰▰▰ ‖ ▰▰▰ Clear ▰▰▰

	PAT-NO	ISSUE-DATE	PATENTEE-NAME	US-CL
☐	T915008	October 1973	Gustavson et al.	364/300
☐	3629843	December 1971	Herbert	364/300

☐	3903399	September 1975	Enns et al.	364/300
☐	4621339	October 1986	Wagner et al.	364/900
☐	4694411	September 1987	Burows	364/578
☐	4694416	September 1987	Wheeler et al.	364/736
☐	4823258	April 1989	Yamazaki	364/200
☐	4862347	August 1989	Rudy	364/200
☐	4888682	December 1989	Ngai et al.	364/200

OTHER PUBLICATIONS

Jacob et al., "Direct-Method Circuit Simulation Using Multiprocessors", Proceedings of the International Symposium on Circuits & Systems, May 1986, pp. 170-173.
Yamamoto et al., "Vectorized LU Decomposition Algorithms for Large Scale Circuit Simulation", IEEE Transactions on Computer Aided Design, vol. Ead-4, No. 3, pp. 232-239, Jul. 1985.

ABSTRACT:

A digital data processing system including a plurality of processors processes a program in parallel to load process data into a two-dimensional matrix having a plurality of matrix entries. So that the processors will not have to synchronize loading of process data into particular locations in the matrix, the matrix has a third dimension defining a plurality of memory locations, with each series of locations along the third dimension being associated with one of the matrix entries. Each processor initially loads preliminary process data into a memory location along the third dimension. After that has been completed, each processor generates process data for an entry of the two-dimensional matrix from the preliminary process data in the locations along the third dimension related thereto. Since the processors separately load preliminary process data into different memory locations, along the third dimension, there is no conflict with accessing of memory locations among the various processors during generation of preliminary process data. Further, since the processors can separately generate process data for different matrix entries from the preliminary data, there is no conflict in accessing of the memory locations among the various processors during of the process data.

10 Claims, 2 Drawing figures

End of Result Set

☐ **Generate Collection** **Print**

L22: Entry 3 of 3   File: USPT   Oct 20, 1992

DOCUMENT-IDENTIFIER: US 5157778 A
** **See image for Certificate of Correction** **
TITLE: Method and apparatus for circuit simulation using parallel processors
including memory arrangements and matrix decomposition synchronization

Detailed Description Text (24):
An heuristic algorithm assigns rows to the slave processors. The first step of this
algorithm is to divide the circuit matrix into blocks of consecutive columns such
that the slave processors can work within blocks without sychronization. The blocks
found by scanning are found by scanning the set of matrix columns from left to
right and assigning them to blocks so that within a block no dependency among
diagonal elements exists when performing LU decomposition. Then in each block rows
containing nonzero subdiagonal elements are assigned to slave processors by
determining the number of updates necessary to complete a row and dividing the
amount of work assigned to the slave processors during the LU decomposition so that
it is balanced among them. For row assignment, the blocks are processed from right
to left.

Current US Cross Reference Classification (2):
712/17

End of Result Set

□ ▇▇▇ Generate Collection ▇▇▇ ‖ Print ‖

L25: Entry 3 of 3   File: USPT   Oct 20, 1992

DOCUMENT-IDENTIFIER: US 5157778 A
** See image for Certificate of Correction **
TITLE: Method and apparatus for circuit simulation using parallel processors
including memory arrangements and matrix decomposition synchronization

Brief Summary Text (12):
Once the matrix is loaded, it is known in the art, using SPICE, to solve the matrix
using sparse matrix LU decomposition. Parallelization of the matrix solution phase
is also important, and presents unique problems. For larger circuits, the CPU time
needed for the matrix solution phase will dominate over that needed for the matrix
load phase. Efficient parallelization schemes are known for full matrices, as is
reported in Thomas, "Using the Butterfly to Solve Simultaneous Linear Equations",
BBN Laboratories Memorandum, March 1985. However, sparse matrices are more
difficult to decompose efficiently in parallel. The LU decomposition algorithm has
a sequential dependency and the amount of concurrent work which can be done at each
step, using SPICE, in a space matrix is small. Algorithms detecting the maximum
parallelism at each step have been proposed for vectorized circuit simulation.
Yamamoto and Takahashi, "Vectorized LU Decomposition Algorithms for Large Scale
Circuit Simulation", IEEE Transactions on computer Aided Design, Vol. Cad-4, No. 3,
pp. 232-239, July 1985. Algorithms based upon a pivot dependency graph and task
queues have been proposed. Jacob et al., supra. The overhead associated with task
queues makes the efficiency of these algorithms questionable.

Brief Summary Text (13):
Recognizing the need for an improved circuit simulation apparatus and method, it is
a general object of present invention to provide a circuit simulation apparatus and
method for simulating LSI and VLSI circuits which eliminates the costly
synchronization requirements of the prior art and, in addition, more efficiently
implements the LU decomposition in parallel using multiple processors.

Detailed Description Text (15):
The tasks performed in parallel by the circuit simulation apparatus and method
according to the present invention are matrix load, matrix LU decomposition and
time step computation.

Detailed Description Text (21):
The present invention employs LU matrix decomposition mathematically similar to
that performed by SPICE. When the LU decomposition requires pivoting, only a single
master processor is used. This is done twice during transient analysis, for full
pivoting, once for the first LU decomposition performed in the DC operating point
computation and second for the first LU decomposition of the transient wave form
computation. For all other decompositions, pivoting is used only relatively
infrequently when a diagonal element used as a pivot is less than a predetermined
threshold value.

Detailed Description Text (22):
During the time the parallel processing is being used for the LU decomposition, the
present invention relates to synchronizing the parallel processors so that they

will perform decomposition operations only on valid data. This is accomplished by assigning each slave processor a set of rows of the circuit matrix. When the SPICE decomposition algorithm progresses to a particular diagonal element, each processor updates the rows in its assigned set. The usual linked list of matrix entries below the diagonal is, according the present invention, broken down into separate lists based upon the rows assigned to each processor. A flag is associated with each diagonal element to ensure that it is never used before its final value is available. Use of this flag results in an efficient synchronization scheme. The flags are shared data accessed by multiple processors in the read mode, but only one in the write mode, so that no locks are needed.

Detailed Description Text (24):
An heuristic algorithm assigns rows to the slave processors. The first step of this algorithm is to divide the circuit matrix into blocks of consecutive columns such that the slave processors can work within blocks without sychronization. The blocks found by scanning are found by scanning the set of matrix columns from left to right and assigning them to blocks so that within a block no dependency among diagonal elements exists when performing LU decomposition. Then in each block rows containing nonzero subdiagonal elements are assigned to slave processors by determining the number of updates necessary to complete a row and dividing the amount of work assigned to the slave processors during the LU decomposition so that it is balanced among them. For row assignment, the blocks are processed from right to left.

Current US Cross Reference Classification (2):
712/17

Other Reference Publication (2):
Yamamoto et al., "Vectorized LU Decomposition Algorithms for Large Scale Circuit Simulation", IEEE Transactions on Computer Aided Design, vol. Ead-4, No. 3, pp. 232-239, Jul. 1985.